

# RPi Node Red: Streaming rpi camera to dashboard

## Goal:

Broadcast a live video feed from the RPi camera to a locally and network accessible webpage.

## Resources:

RPi and a RPi camera.

## Hardware:

Only setup here is connecting the RPi camera to the pi using a ribbon cable.

## Installing Streaming Software:

We'll be following this tutorial here:  
<https://elinux.org/RPi-Cam-Web-Interface>

The page there is extremely verbose and scary, but the actual setup is very simple and should only take a few minutes.

First open a terminal, and enter

```
sudo apt-get update
```

This will update your repository so the right links are there.

Then paste / type this into the same terminal and press enter.

```
git clone https://github.com/silvanmelchior/RPi_Cam_Web_Interface.git
```

This will start downloading the git repo for the web interface; this should only take a few seconds.

```
cd RPi_Cam_Web_Interface
```

Enters the directory that we just downloaded.

```
./install.sh
```

runs the script to start installing everything, it will prompt you for some settings, press enter to use the defaults.

In the article we're following it mentions

The scripts are

- install.sh main installation as used in step 4 above
- update.sh check for updates and then run main installation
- start.sh starts the software. If already running it restarts.
- stop.sh stops the software
- remove.sh removes the software
- debug.sh is same as start but allows raspimjpeg output to console for debugging

To run these scripts make sure you are in the RPi\_Cam\_Web\_Interface folder then precede the script with a ./

- E.g. To update an existing installation ./update.sh
- E.g. To start the camera software ./start.sh
- E.g. To stop the camera software ./stop.sh

We just ran the 'install.sh', which installs everything. Now if we want to start the stream we'll use 'start.sh'

```
./start.sh
```

If we want to start the stream, say after restarting the pi we'll have to navigate back to this directory and run the start script.

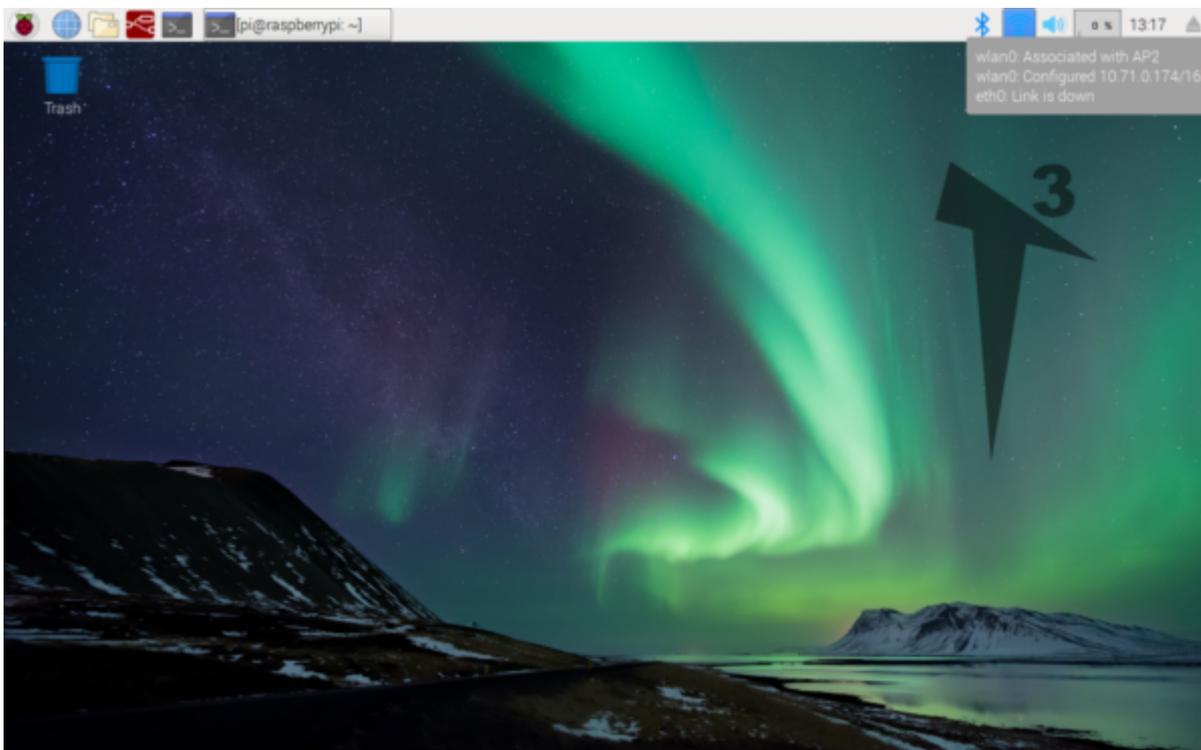
That process just looks like this:

```
cd RPi_Cam_Web_Interface  
./start.sh
```

## Viewing the stream:

Now for the fun part, actually viewing the stream.

First we'll need our ip address; this can be found by hovering the mouse over the WiFi applet like this:

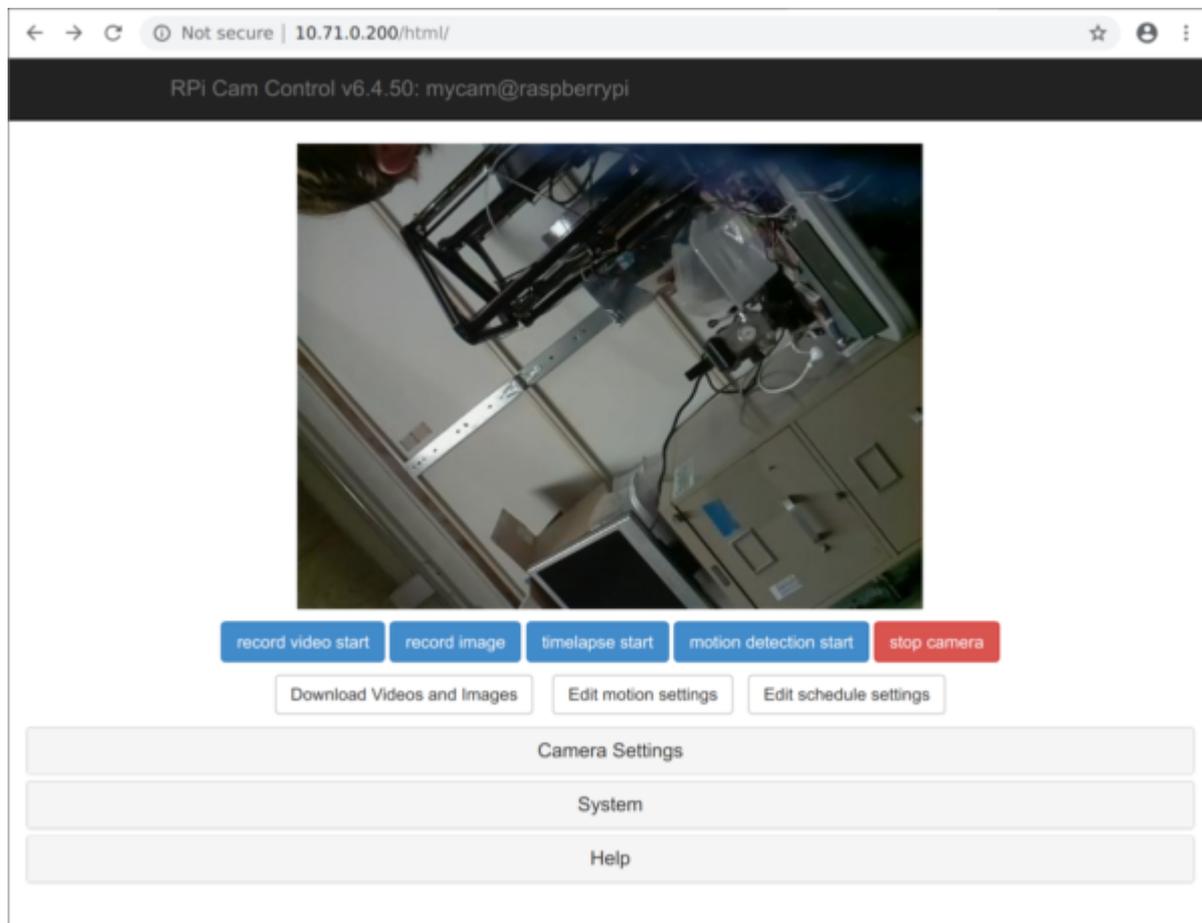


Next, on the same pi or any computer on the same local network we can open the stream via this address.

The url will look like this, except with the ip address switched out for your own.

[10.71.0.174/html](http://10.71.0.174/html)

Here's what the default page looks like when we're streaming.



You can edit all kinds of settings here, that aren't really necessary for a basic setup.

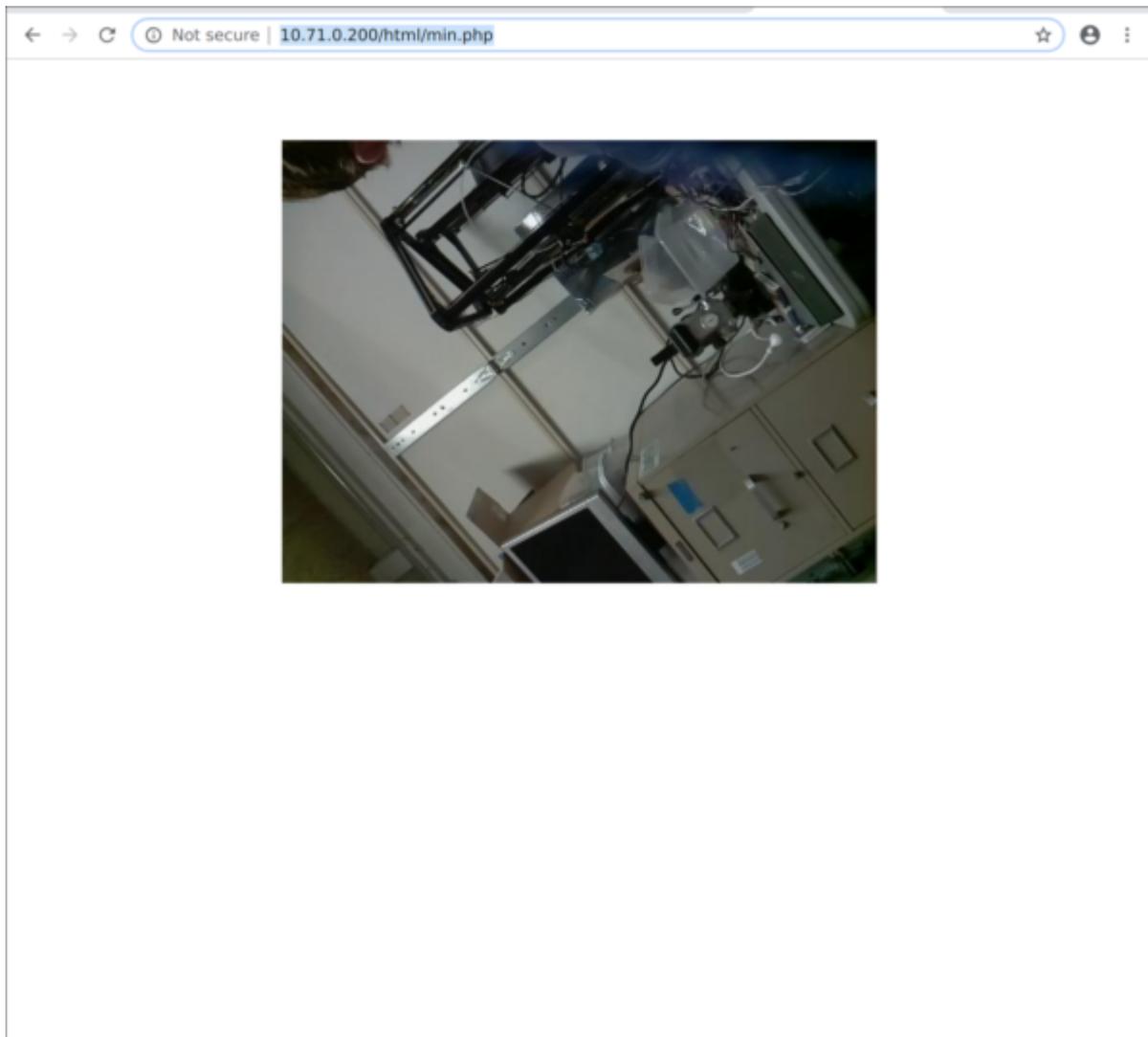
If your image happens to be upside down you can change the rotation in the "Camera Settings" tab here:

Exposure mode, default 'auto':	Auto ▾
White Balance, default 'auto':	Auto ▾
White Balance Gains (x100):	gain_r 150 gain_b 150 OK
Image Effect, default 'none':	None ▾
Colour Effect, default 'disabled':	Disabled ▾ u:v = 128 : 128 OK
Image Statistics, default 'Off':	Off ▾
Rotation, default 0:	Rotate_180 ▾
Flip, default 'none':	Both ▾
Sensor Region, default 0/0/65536/65536:	x: 0 y: 0 w: 65536 h: 65536 OK
Shutter speed (0...330000), default 0:	0 OK
Image quality (0...100), default 10:	10 OK

There's also a more minimal page that only shows the video here:

<http://10.71.0.184/html/min.php>

Remember that the IP address will be different for you.



# Embedding the stream into node-red dashboard:

The T3 RPi kit comes with the node-red dashboard nodes installed already, and this is what we'll use to view the stream in node-red. The advantage to doing that is that you can have buttons, graphs, or other data alongside the video feed.

Here is an example where we have the feed coming from a remote controlled car, with buttons alongside to drive the car around.



FWD

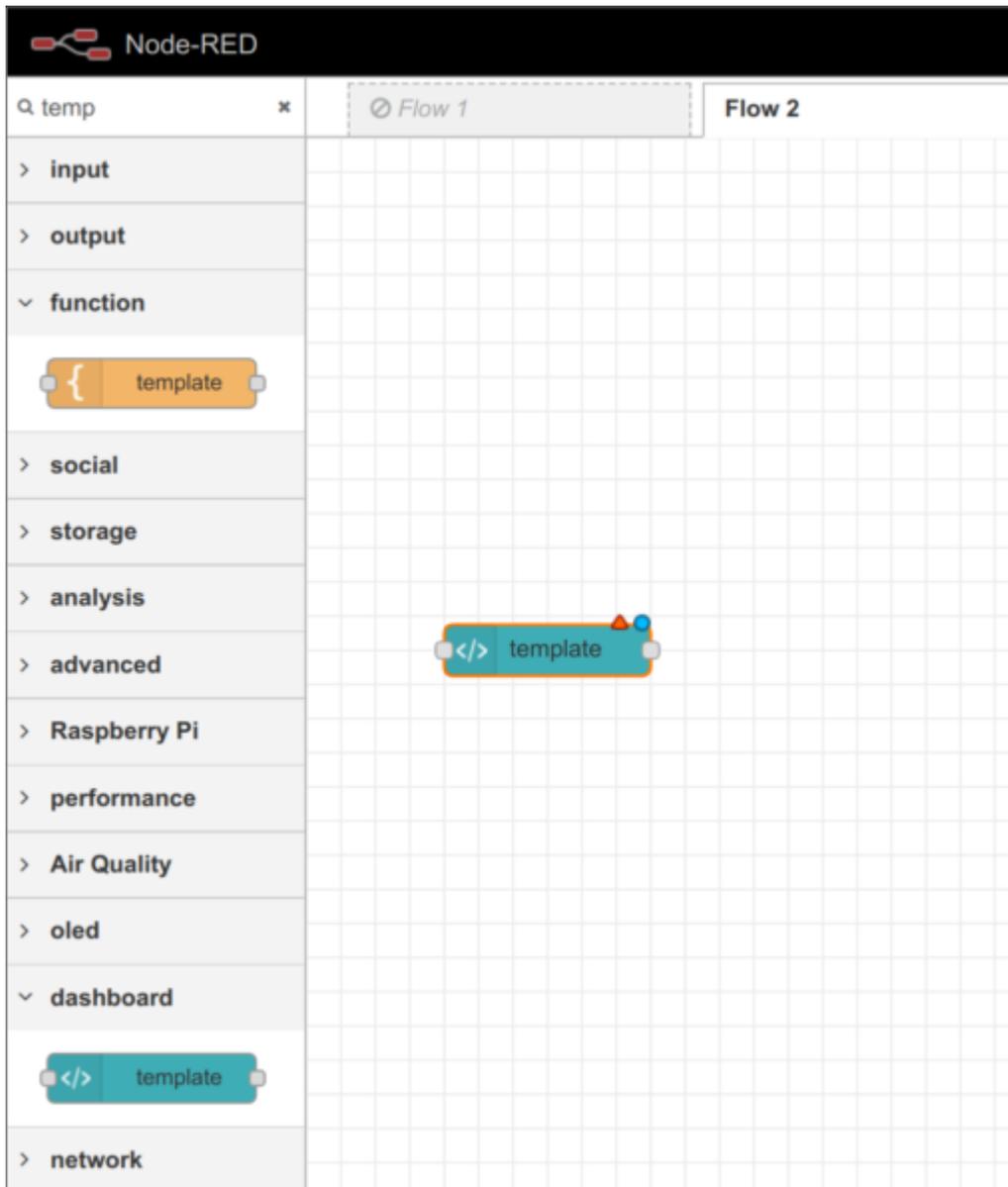
BACK

STOP

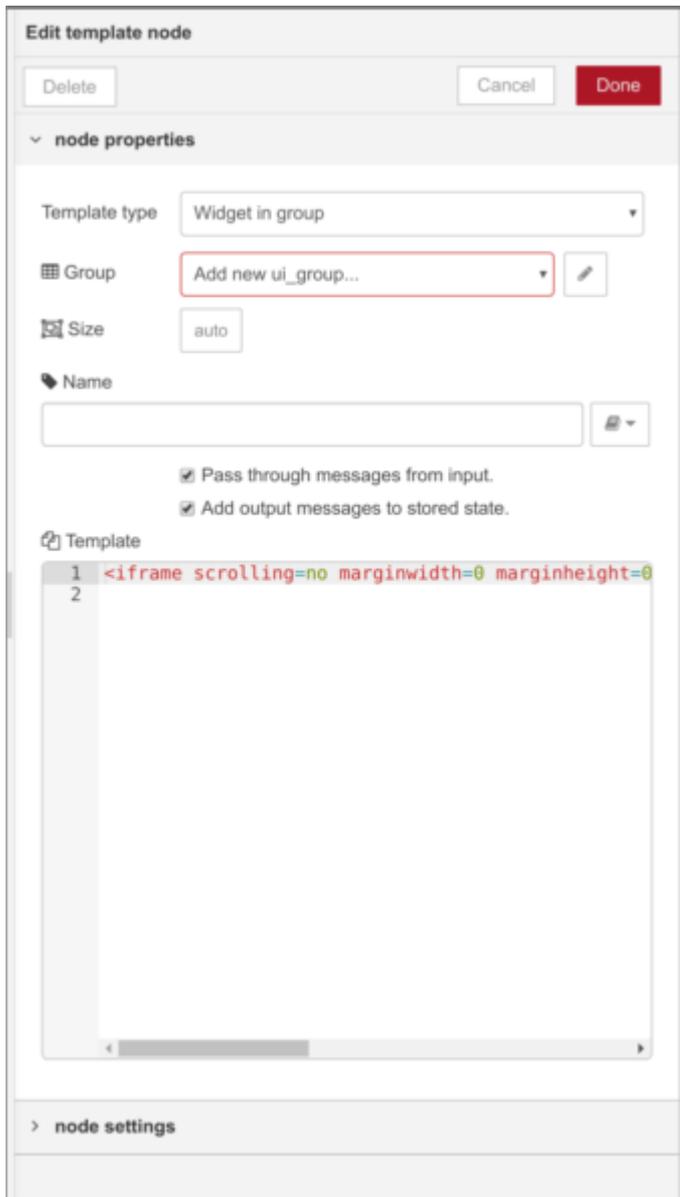
RIGHT

LEFT

In this tutorial we'll focus on getting the video itself to display.



Here is what the configuration for the template node looks like:



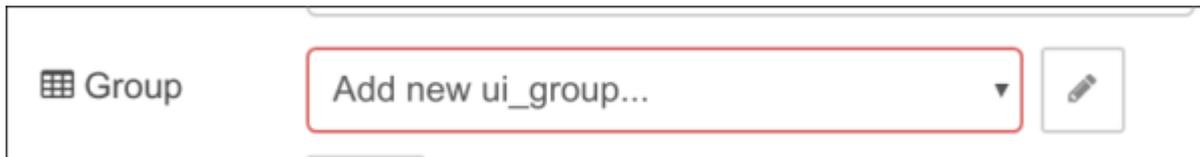
everything is set to default, except we've added some html code to the template box.

Here's the code:

```
<iframe scrolling=no marginwidth=0 marginheight=0
frameborder=0 height=439 width=553
src="http://10.71.0.200/html/min.php"></iframe>
```

For it to work on your template node, you'll have to replace the url here with the one for your pi. So the IP address will be different.

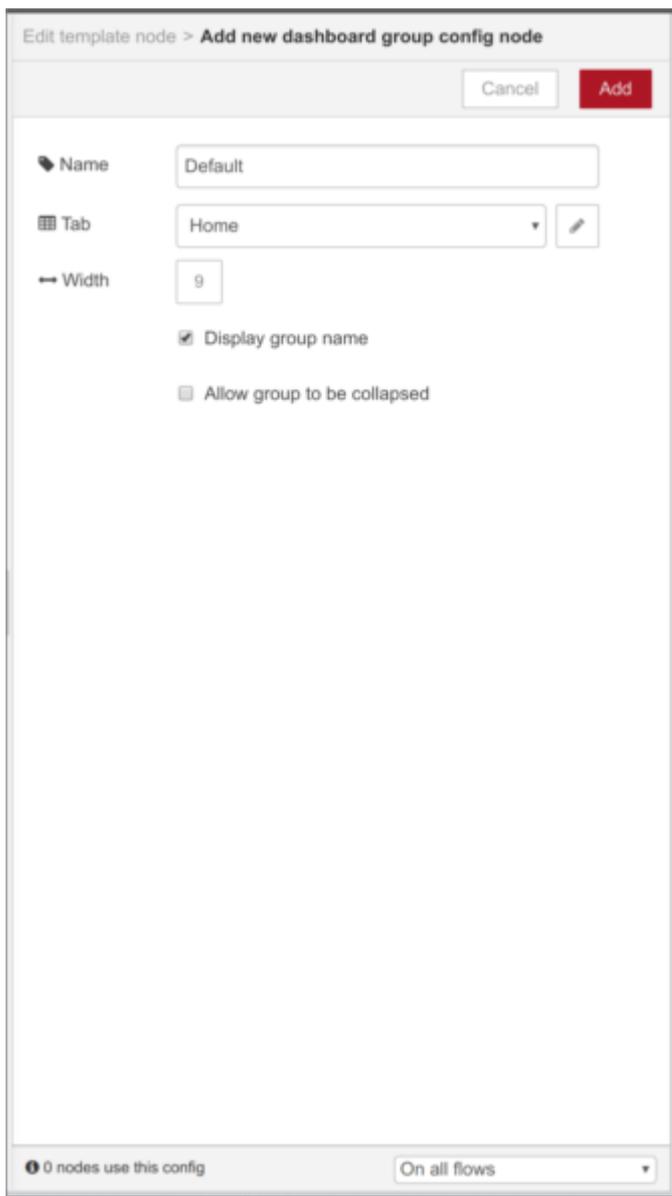
The other change is adding a new ui\_group.



If you click the pen here, it'll open the panel to define the new default group.

You can leave this all at default, but I think it looks better with the video if you raise the width slightly.

Here's mine with the width raised to 9



Now if you deploy and navigate to [http://your\\_ip:1880/ui](http://your_ip:1880/ui) you should be able to see your video stream embedded in the node-red dashboard.

Default



You can fine tune the iframe settings like width and height in the template node, and the layout size in the dashboard options.